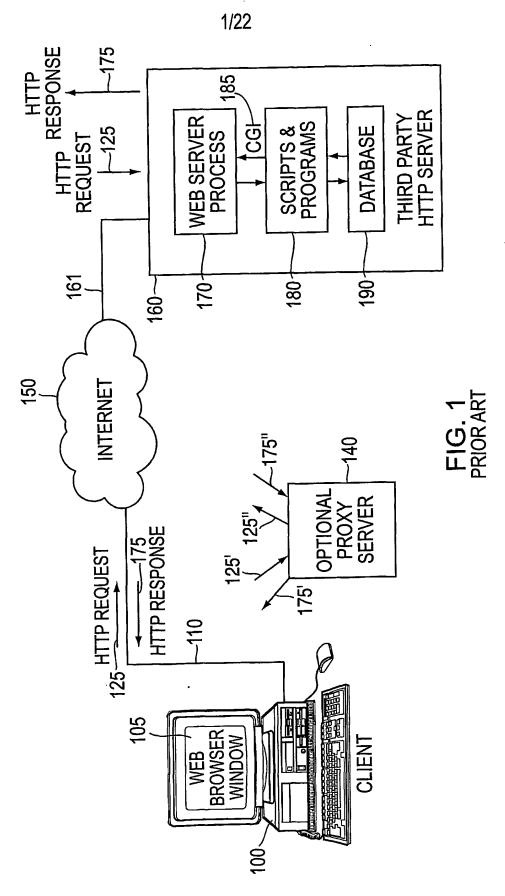
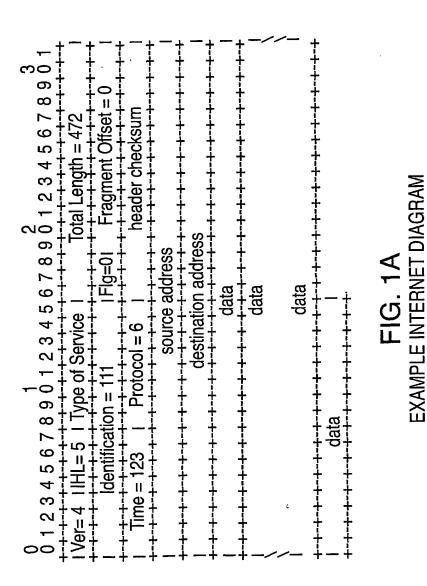
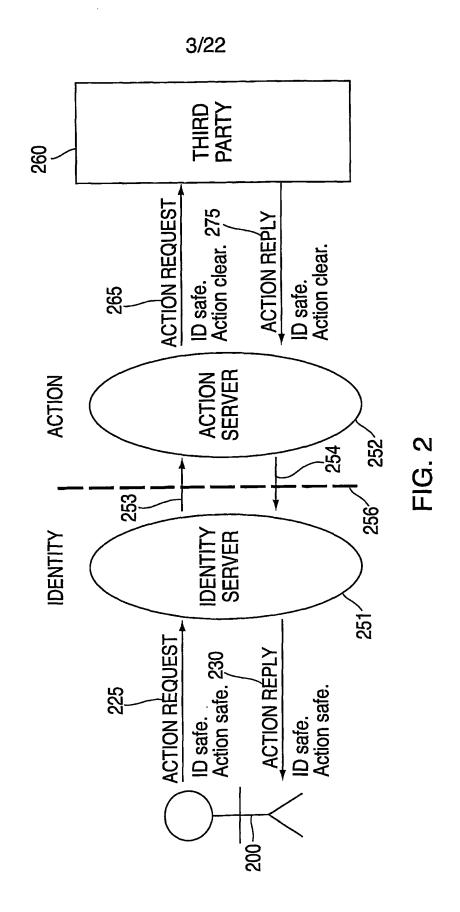
PCT/US02/08275



SUBSTITUTE SHEET (RULE 26)

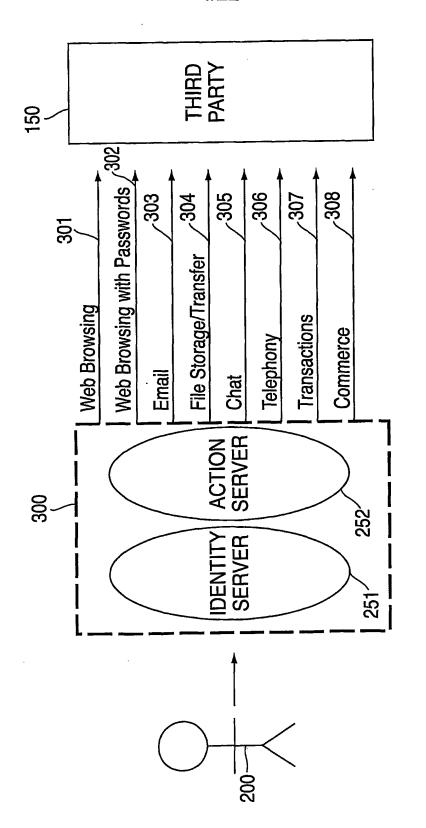




**SUBSTITUTE SHEET (RULE 26)** 

FIG. 3





**SUBSTITUTE SHEET (RULE 26)** 

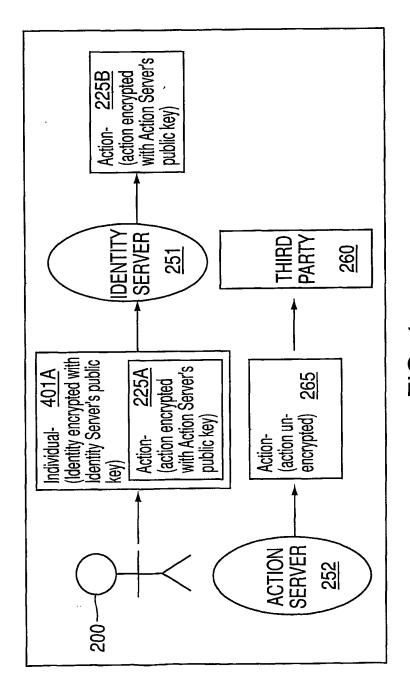


FIG. 4

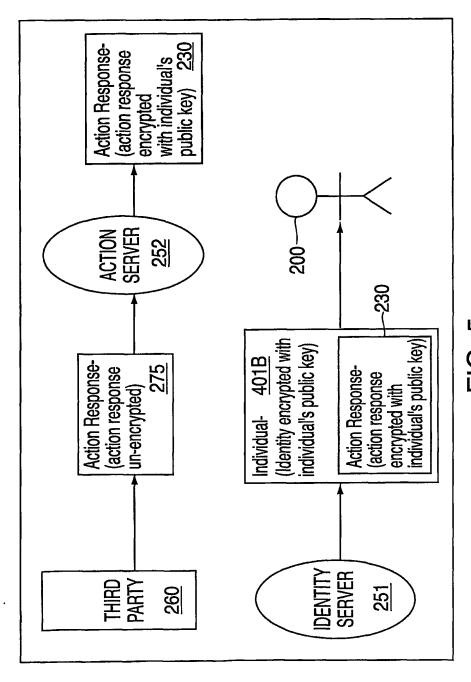
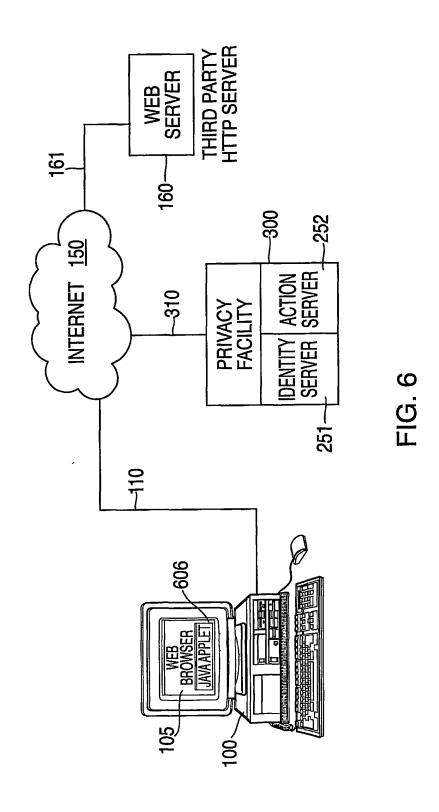


FIG. 5



**SUBSTITUTE SHEET (RULE 26)** 

WO 02/095545 PCT/US02/08275

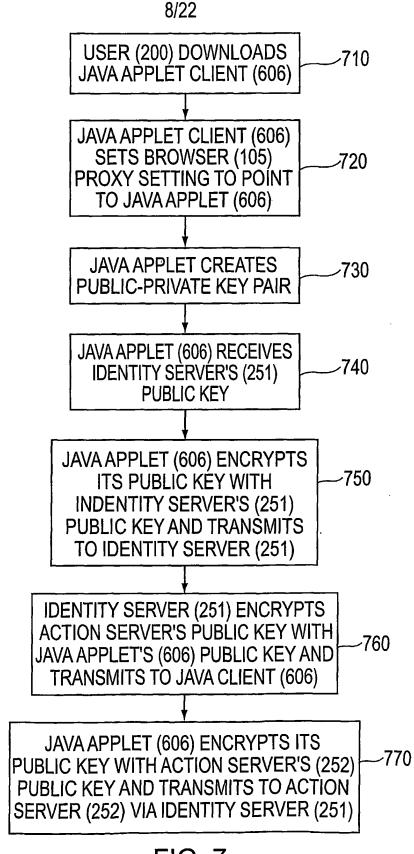
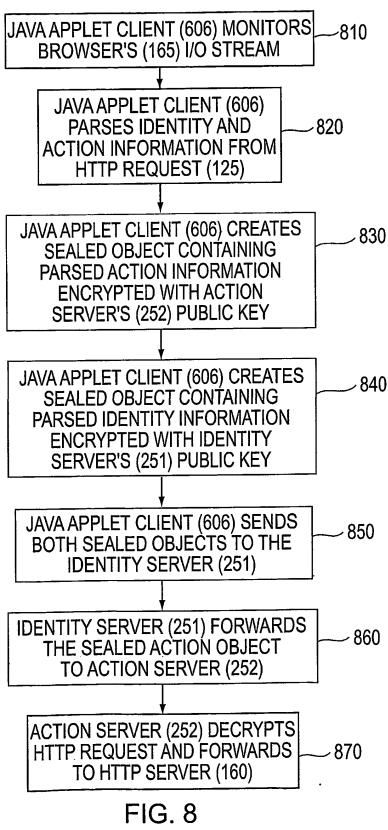


FIG. 7
SESSION INITIALIZATION

PCT/US02/08275





REQUEST TRANSMISSION

## 10/22

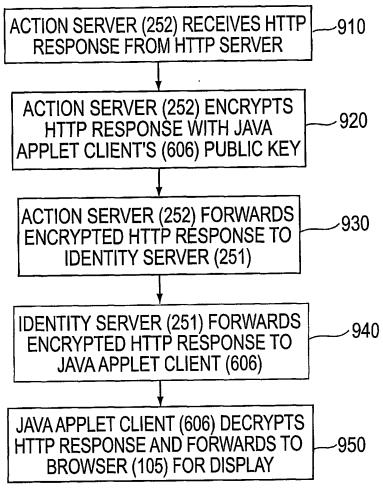


FIG. 9
RESPONSE TRANSMISSION

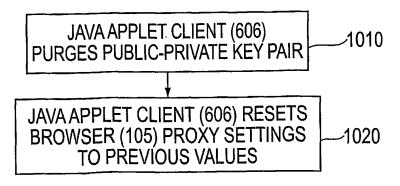
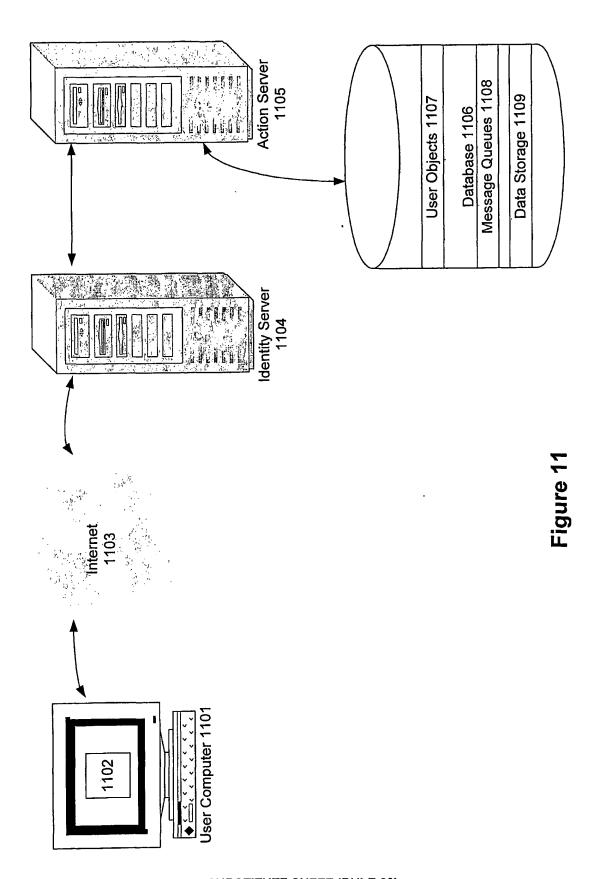


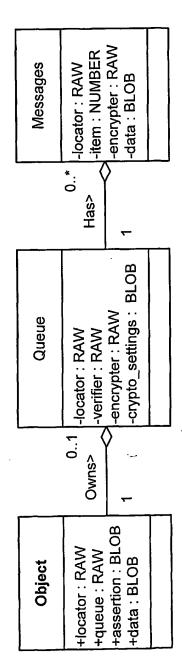
FIG. 10
SESSION TERMINATION
SUBSTITUTE SHEET (RULE 26)



12/22

System\_Values
-id : VARCHAR2
-value : RAW

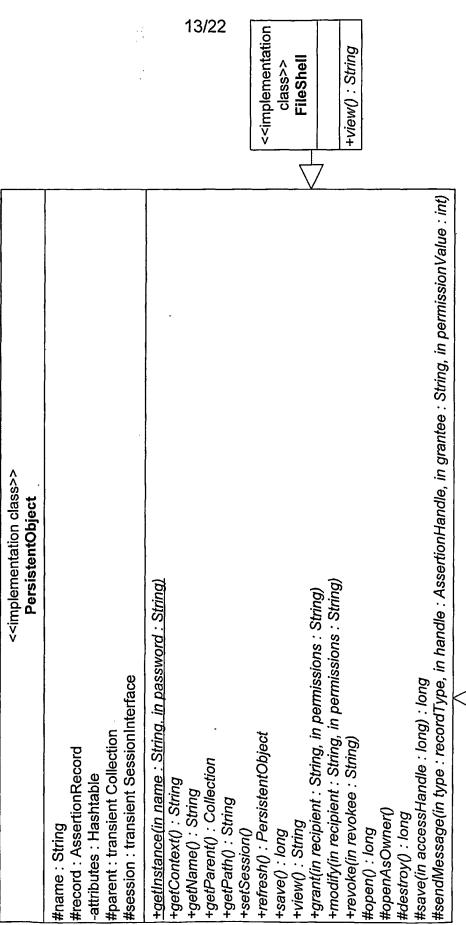
Figure 12

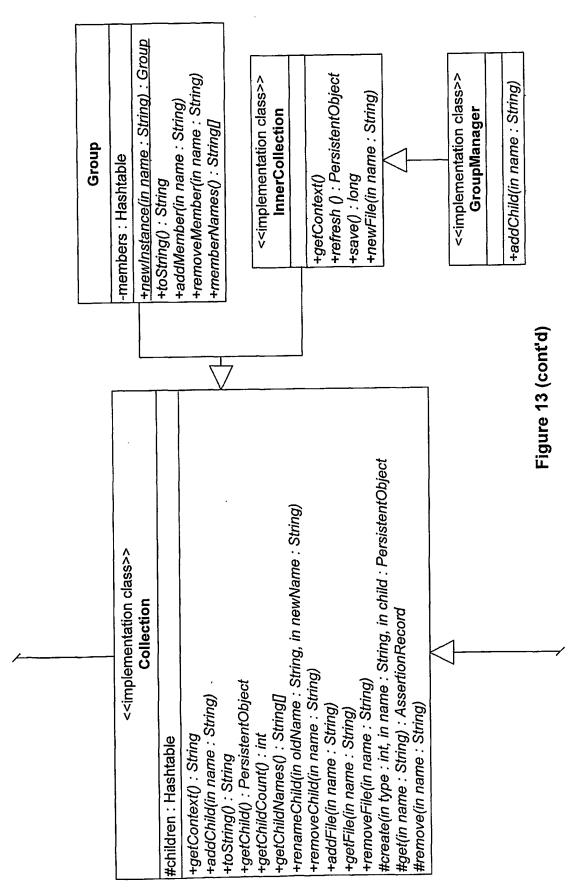


Logical Data Model

Figure 13

Client Data Objects





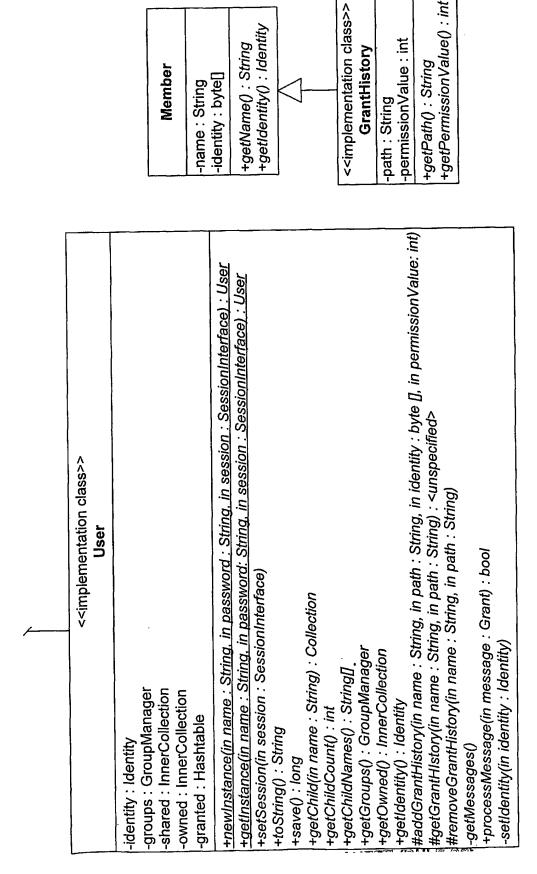


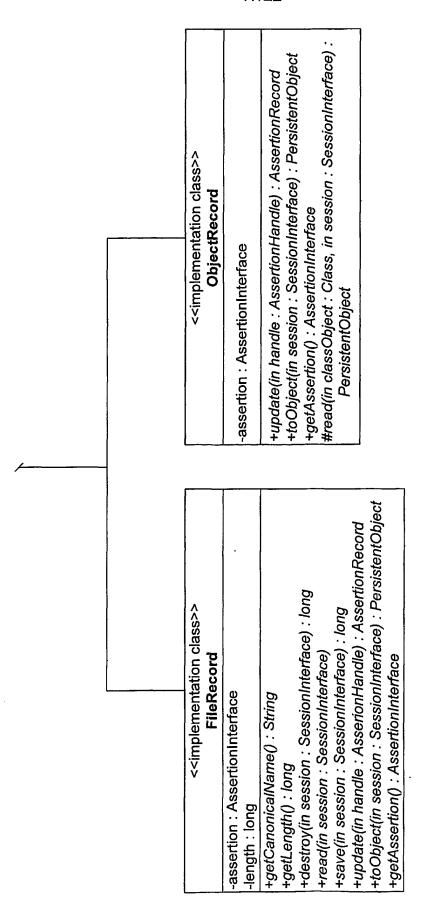
Figure 13 (cont'd)

Figure 14

Client Data Record Objects

| < <implementation class="">&gt;</implementation>   |
|--|
| -name: String -type: record Type -decrypter: byte[] -encrypter: byte[] -settings: CryptoSettings   |
| +getName(): String<br>+setName(in name: String)<br>+getType(): recordType<br>+getCypto(): CryptoSettings<br>+setCypto(in settings: CryptoSettings)<br>+setCypto(in settings: CryptoSettings)<br>+toString(): String<br>+getAttribute(): String<br>+getAttribute(in name: String)<br>+setAttribute(in name: String)<br>+removeAttribute(in name: String)<br>+removeAttribute(in name: String)<br>+setStream(in out: OutputStream, in length: long, in session: SessionInterface)<br>+setStream(in in: InputStream, in length: long, in session: SessionInterface)<br>+setStream(in handle: AssertionHandle): AssertionRecord {abstract}<br>+toObject(in session: SessionInterface): PersistentObject {abstract}<br>+getAssertion(): AssertionInterface {abstract} |





18/22

| class>>  |          |
|--|----------|
| < <implementation< td=""><td>Identity</td></implementation<> | Identity |

-messageDecrypter : byte]]

-queueldentifier : byte[]

-ownerldentifier : byte[]

-crypto: transient AsymmetricCryptoPackage -ownerVerifier : byte[]

+assertOwnership(in cryptography : CryptoManager, in handle : ChallengeHandle) : Challenge +getOwnerVerifier() : byte[]

+assertMembership(in cryptography : CryptoManager, in handle : ChallengeHandle, in identity : byte[]) : Challenge +assertAuthorship(in cryptography : CryptoManager, in handle : ChallengeHandle) : Challenge

+decryptMessage(in cryptography : CryptoManager, in messageKey : byte∏, in settings : CryptoSettings, in messageBody : byte∏ : byte∏

<<enumeration>> recordType

Figure 14 (cont'd)

+getPermissionValue(): int +getDescription(): String

+toString(): String

+getType(): recordType permissionValue : int

-description : String

-grantor: String

-type: recordType

+getGrantor(): String

<<implementation class>>

Grant

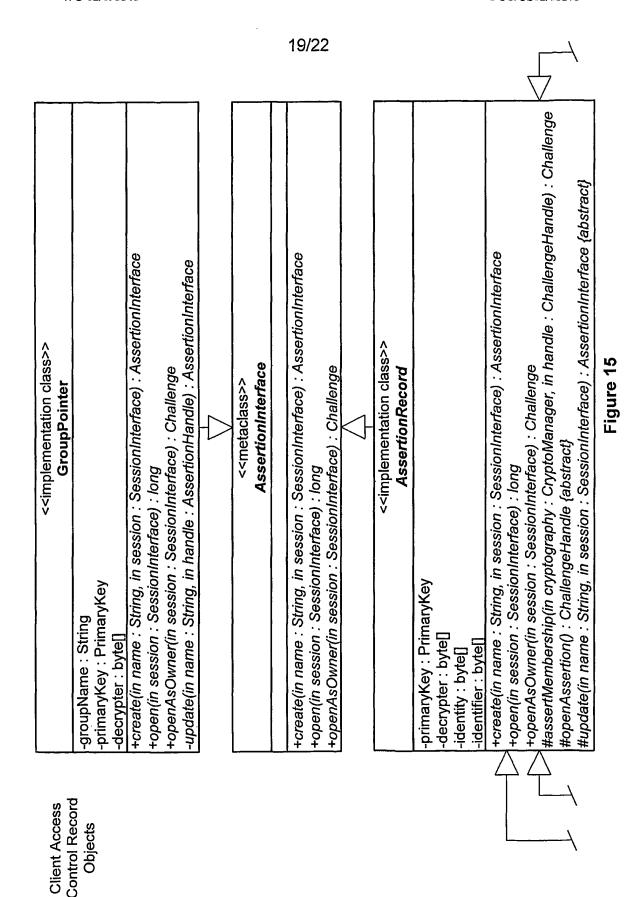


Figure 15 (Cont'd)

| , | -groupName : String  | <del></del> |
|---|--|-------------|
|   | +newInstance(in name : String, in session : SessionInterface) : GroupRecord<br>#openAssertion() : ChallengeHandle<br>#update(in name : String, in session : SessionInterface) : AssertionInterface   | <del></del> |
|   | AccessRecord   |             |
|   | +newInstance(in primaryKey: PrimaryKey, in decrypter: byte[], in identity: byte[], in identifier: byte[]) #open(in decrypter: byte[], in session: SessionInterface) #openAssertion(): ChallengeHandle #update(in name: String, in session: SessionInterface): AssertionInterface |             |
|   | UserRecord   |             |
|   | -keyACL : byte[]<br>-keyData : byte[]  | <del></del> |
|   | -identifier: byte[]<br>-identifier: byte[]   |             |
|   | +newInstance(in name: String, in password: String, in session: SessionInterface): UserRecord   | <del></del> |
|   | +create(in name : String, in session : SessionInterface) : AssertionInterface<br>+open(in session : SessionInterface) : Iona   |             |
|   | +openAsOwner(in session : SessionInterface) : Challenge<br>+update(in session : SessionInterface) : DataRecord   | ]           |
|   | +setStream(in serialized : byte[], in session : SessionInterface) : long<br>+toObject(in session : SessionInterface) : PersistentObject  |             |
|   | +toRecord(in name : String, in type : recordType) : UserRecord   |             |

| Objects  |
|----------|
| Control  |
| Access   |
| Server / |

| < <implementation class="">&gt; AccessControlList</implementation>  |
|---|
|   |
| +newinstance(in client : DatabaseClient, in primaryKey : PrimaryKey, in encrypter : byte[], in ownerVerifier : byte[], in identity : byte[], in permissionValue : int) : AccessControll ist |
| +newinstance(in client : DatabaseClient, in primaryKey : PrimaryKey, in encrypter : byteff, in parent : Group, in groupIdentity :   |
| bytefl. in groupVerifier: byteff): AccessControlList  |
| +newinstance(in client : DatabaseClient, in parent : AccessControlList, in primaryKey : PrimaryKey, in encrypter : byte[]) :  |

+getInstance(in client : DatabaseClient, in primaryKey : PrimaryKey, in decrypter : byteff) : AccessControlList **AccessControlList** 

+assert(in permissionType : int, in identity : byte[]) : int

+addEntry(in identity : byte∏, in verifier : byte∏, in permissionValue : int) +setEntry(in identity : byte∏, in verifier : byte∏, in permissionValue : int)

+removeEntry(in identity : byte∏, in verifier : byte∭

-checkPermissions(in permissionType : int, in identity : byte[]) : int

<<implementation class>>

Group

-accessIdentity: byte[ -accessVerifier : byte[] +newinstance(in client : DatabaseClient, in pk : PrimaryKey, in encrypter : byte[], in ownerVerifier : byte[], in accessIdentity : byte[], in accessVerifier: byte[], in identity: byte[], in verifier: byte[]): Group

+getInstance(in client:DatabaseClient, in primaryKey:PrimaryKey, in decrypter:byte[]):Group

+getAccessIdentity() : byte[]

+getAccessVerifier() : byte∏

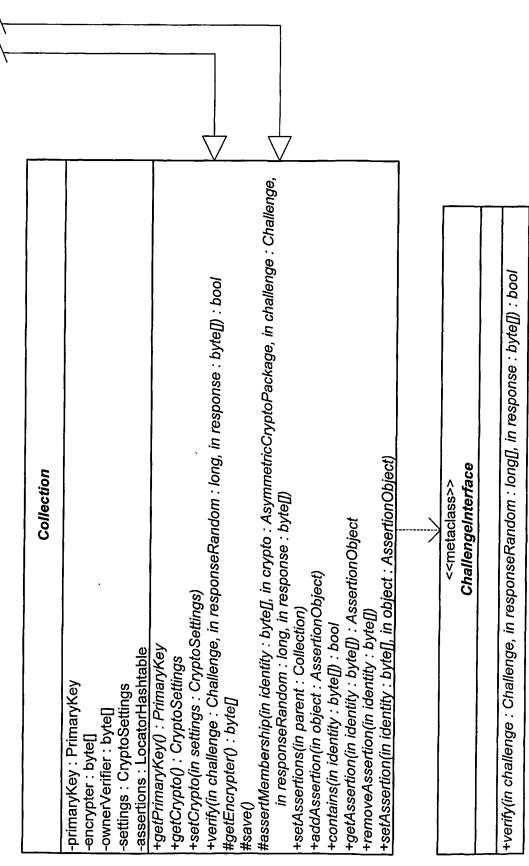
+destroy()

+addMember(in identity∶byte∏, in verifier∶byte∬

+removeMember(in identity : byte[])

Figure 16

Figure 16 (cont'd)



SUBSTITUTE SHEET (RULE 26)